CLAIMS:

1. A method for accommodating a legacy application, the method comprising:

obtaining a request for a high-level credential from a legacy application;

marshalling the requested credential;

returning the marshaled credential to the application.

2. A method as recited in claim 1 further comprising, after the obtaining, seeking the requested credential in a database of credentials.

3. A method as recited in claim 1, wherein a high-level credential is a credential selected from a group composed of X.509 Certificates and bio-metrics.

4. A method as recited in claim 1, wherein the marshaled credentials appear to be a conventional username/password pair to the legacy application.

5. A method as recited in claim 1, wherein marshalling comprises:

obtaining the requested high-level credential;

pickling the requested high-level credential to generate a low-level credential that represents the requested high-level credential while appearing to be a conventional username/password pair to the legacy application.

6. A method as recited in claim 1, wherein the legacy application never has access to the high-level credential.

7. A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method as recited in claim 1.

8. In a computing environment where processes have a provision for low-level credentials but have no provision for high-level credentials, a method for accommodating such processes comprising:

obtaining a request for a credential from a process, wherein the requested credential is a high-level credential;

retrieving the requested credential from a database;

converting the requested high-level credential into a format approximating a low-level credential and representative of the requested high-level credential;

returning the converted credential to the process.

9. A method as recited in claim 8, wherein a high-level credential is a credential selected from a group composed of X.509 Certificates and bio-metrics.

10. A method as recited in claim 8, wherein the converted credentials appear to be a conventional username/password pair to the process.

11. A method as recited in claim 8, wherein the process never has access to the high-level credential.

12. A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method as recited in claim 8.

13. A method for authenticating a user to a network, the method comprising:

obtaining a request for a credential to authenticate the user to access a resource within the network, wherein the resource requires an appropriate credential before the user may access the resource;

locating the appropriate credential;

returning the appropriate credential to the resource within the network, so that the resource allows the user to access such resource;

wherein the obtaining, locating, and returning are performed without user interaction so that the user need not be aware that such steps are being performed.

14. A method as recited in claim 13 further comprising repeating the obtaining, locating, and returning for a different network that is authenticated using a different credential.

15. A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method as recited in claim 13.

16. A method for concurrently accessing a first resource on a first network and a second resource on a second network, the method comprising:

first obtaining a first request for a first credential to authenticate a user to access a first resource of the first network, wherein the first resource requires an appropriate first credential before the user may access the first resource;

first locating the appropriate first credential;

first returning the appropriate first credential to the first resource of the first network, so that the first resource allows the user to access the first resource;

wherein the first obtaining, locating, and returning are performed without user interaction so that the user need not be aware that such steps are being performed;

second obtaining a second request for a second credential to authenticate a user to access a second resource of the second network, wherein the second resource requires an appropriate second credential before the user may access the second resource;

second locating the appropriate second credential;

second returning the appropriate second credential to the second resource of the second network, so that the second resource allows the user to access the second resource;

wherein the second obtaining, locating, and returning are performed without user interaction so that the user need not be aware that such steps are being performed

17. A computer-readable medium having computer-executable instructions that, when executed by a computer, performs the method as recited in claim 16.

18. A credential management architecture, comprising:

a trusted computing base (TCB) that has full access to persisted credentials, the TCB being configured to interact with an untrusted computing layer (UTCL) that accesses the persisted credentials via the TCB;

the TCB comprises:

a credential management module configured to receive requests from the UTCL for a credential for a resource, the credential being associated with a user;

a credential database associated with the user, wherein credentials are persisted within the database;

the credential management module being configured to retrieve credentials from the database.

19. An architecture as recited in claim 18, wherein credential management module is further configured to marshal a requested credential and return the marshaled credential to the UTCL.

20. An architecture as recited in claim 18, wherein the marshaled credentials appear to be a conventional username/password pair to the UTCL.

21. A computer-readable medium having computer-executable instructions that, when executed by a computer, employ an architecture as recited in claim 18.

22. An operating system embodied on a computer-readable medium having computer-executable instructions that, when executed by a computer, employ an architecture as recited in claim 18.

23. An apparatus comprising:

a processor;

a marshaler executable on the processor to:

obtain a high-level credential;

convert the high-level credential to generate a representation of the high-level credential that is formatted as a low-level credential so that it appears to be a conventional username/password pair.

24. A low-level-credential-application accommodation system comprising:

a request obtainer configured to obtain a request for a high-level credential from a low-level-credential-application;

a credential retriever configured to retrieve the requested credential from a database of credentials;

a marshaller configured to marshal the requested credential and return the marshaled credential to the low-level-credential-application.

25. A system as recited in claim 24, wherein a high-level credential is a credential selected from a group composed of X.509 Certificates and bio-metrics.

26. A system as recited in claim 24, wherein the marshaled credentials appear to be a conventional username/password pair to the legacy application.

27. A system as recited in claim 24, wherein marshaller is further configured to convert the requested high-level credential to generate a low-level credential that represents the requested high-level credential while appearing to be a conventional username/password pair to the low-level-credential-application.

28. A system as recited in claim 24, wherein the legacy application never has access to the high-level credential.

29. A system for authenticating a user to a network, the system comprising:

a request obtainer configured to obtain a request for a credential to authenticate the user to access a resource within the network, wherein the resource requires an appropriate credential before the user may access the resource;

a credential retriever configured to retrieve the appropriate credential from a database of credentials;

a credential returner configured to return the appropriate credential to the resource within the network, so that the resource allows the user to access such resource;

wherein the obtainer, retriever, and returner are further configured to operate without user interaction.

30. An operating system comprising a system as recited in claim 29.

31. A network environment comprising a system as recited in claim 29.

32. An application programming interface (API) method comprising:

receiving a CredUI-promptfor-credentials call having a set of parameters comprising a TargetName, Context, AuthFlags, and Flags;

parsing the call to retrieve the parameters to determine a specified resource;

obtaining a credential;

associating the credential with the specified resource;

persisting the credential into a database while maintaining the credential's association with the specified resource.

33. A method as recited in claim 32, wherein the set of parameters further comprises an indicator of a data structure containing customized information to display in conjunction with a user interface.

34. An application programming interface (API) method comprising:

receiving a CredUI-promptfor-credentials call having a set of parameters comprising a TargetName, UserName, Password, and Flags;

parsing the call to retrieve the parameters to determine a requesting application;

obtaining a low-level credential from a user, wherein such credential includes a username and a password;

returning the low-level credential to the requesting application.

35. A method as recited in claim 34, wherein the set of parameters further comprises an indicator of a data structure containing customized information to display in conjunction with a user interface.